

VENTURUS 4 TECH



Desafio HTML:

[github.com/adrianomaringolo/
4tech2019-react](https://github.com/adrianomaringolo/4tech2019-react)

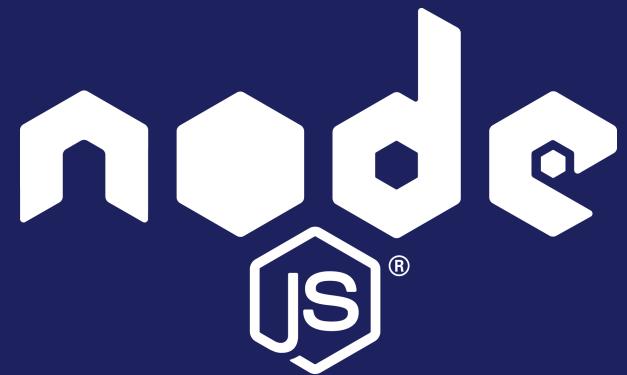
```
5  abort("The Rails environment is running in production mode! Please run\n6  require 'spec_helper'\n7  require 'rspec/rails'\n8\n9  require 'capybara/rspec'\n10 require 'capybara/rails'\n11\n12 Capybara.javascript_driver = :webkit\n13 Category.delete_all; Category.create!\n14 Shoulda::Matchers.configure do |config|\n15   config.integrate do |with|\n16     with.library :rails\n17   end\n18 end\n19\n20 # Add additional configuration here if required\n21\n22 # Requires supporting files within the same directory as spec/\n23 # run as spec files by default\n24 # in _spec.rb will both be required.\n25 # run twice. It is recommended to only do this for once\n26 # end with _spec.rb. You can have multiple\n27 # options on the configuration hash.\n28 # option on the configuration hash.\n29 # mongoid"
```

Material do curso:

bit.ly/vnt4tech19

```
5  abort("The Rails environment is running in production mode! Please run\n6  require 'spec_helper'\n7  require 'rspec/rails'\n8\n9  require 'capybara/rspec'\n10 require 'capybara/rails'\n11\n12 Capybara.javascript_driver = :webkit\n13 Category.delete_all; Category.create\n14 Shoulda::Matchers.configure do |config|\n15   config.integrate do |t|\n16     t.library 'rails', '4.2.7'\n17     t.with_library 'mongoid', '4.0.1'\n18     t.default_with_library 'mongoid'\n19   end\n20 end\n21\n22 # Add additional requires below this line if you need them\n23\n24 # Requires supporting files within the same directory as\n25 # this file or explicitly\n26 # specify a path.\n27 # in _spec.rb will both be required by\n28 # run twice. It is recommended to\n# run with _spec.rb. You can\n# caution on the command line with\n# \"rspec --require Mongoid\"
```

VENTURUS4TECH





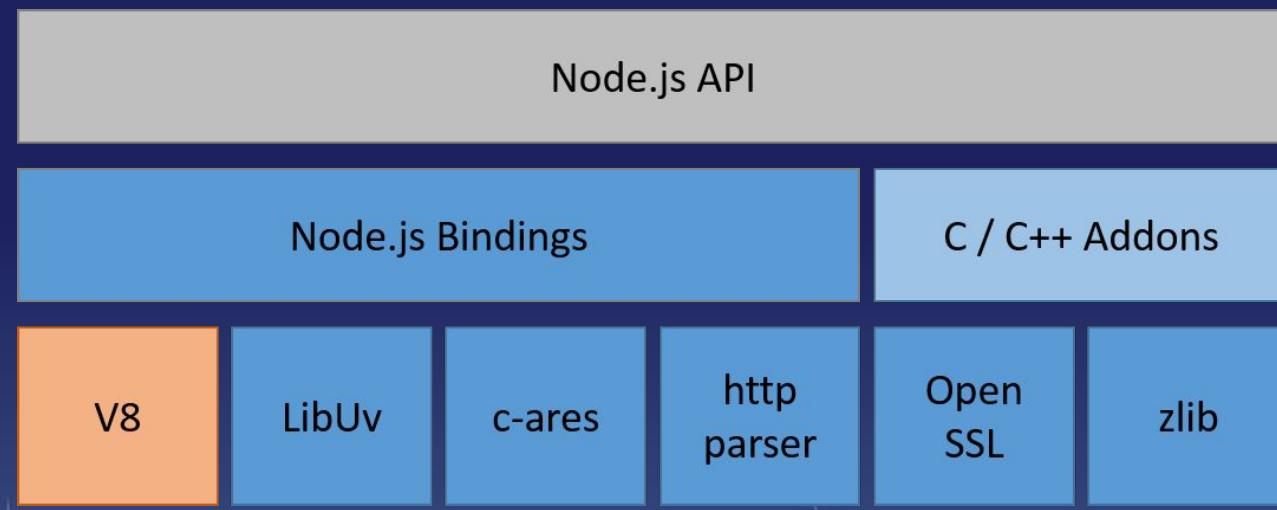
O que é Node.js?

- Plataforma para criação e execução de servidores para aplicações web
- Escrito em C++ (performance)
- Open-source (código aberto)
- Cross-platform (Linux, Mac, Windows)



Saiba mais: [Site oficial do Node.js](https://nodejs.org/)

- Por baixo dos panos o Node.js usa o interpretador de javascript **V8** criado pelo **Google** para o **Chrome**, além, claro, de outras funcionalidades:





O que não é?

- Uma linguagem de programação
- Um framework
- Multi-thread
 - ◆ **Simplifica** muito a construção de uma aplicação.
 - ◆ Imperceptível na maioria dos casos por não ser **blockante**.

Não é um Framework?





Diferença entre Framework e uma Biblioteca:

<https://www.c-sharpcorner.com/UploadFile/a85b23/framework-vs-library/>

Framework

"... todo o fluxo de controle já está nele e existem diversos pontos predefinidos em branco que devemos preencher com nosso código..."

... define a arquitetura geral de um sistema"

Biblioteca

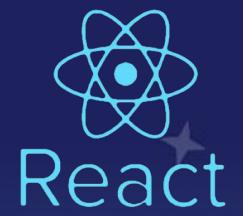
"... é apenas uma coleção de definições de classes..."

"... o seu objetivo é o reuso de código, em outras palavras, você simplesmente faz uso do código que já foi escrito por outros desenvolvedores..."

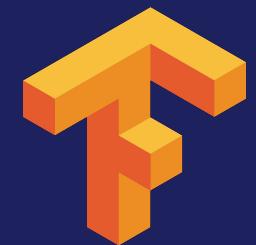
Frameworks



...



...



Bibliotecas



Como nossa aplicação vai funcionar?

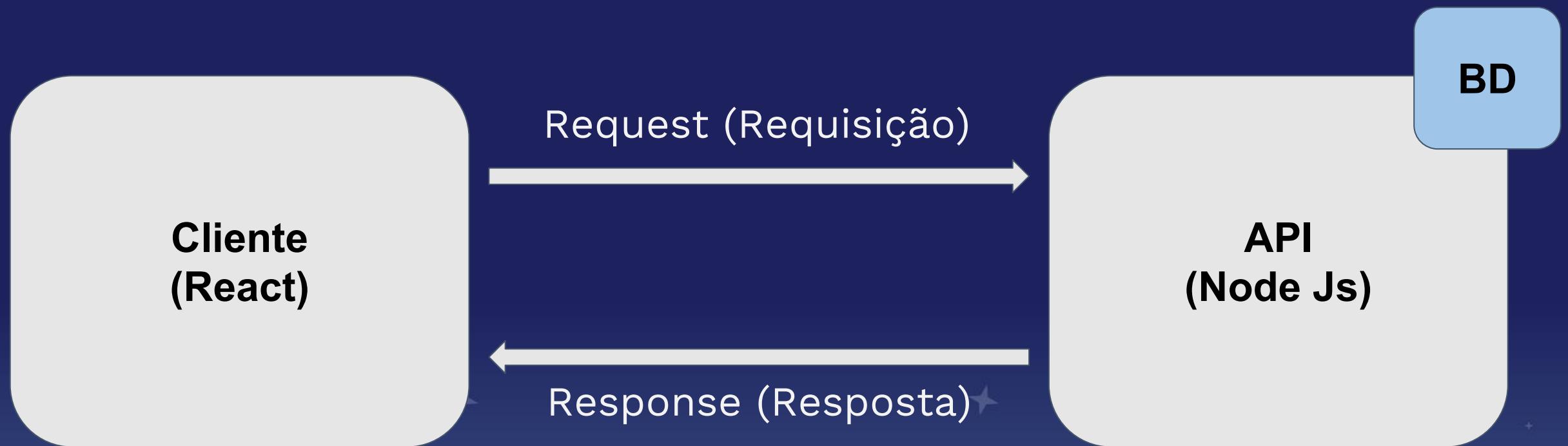


Pede / Requisita / Ordena

Responde



REST API Design



Requisições HTTP(s)

Requisição

```
GET /HTTP/1.1
Host: www.google.com.br
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:29.0) Gecko/20100101 Firefox/29.0
Accept: text/html,application/xhtml+xml,application/xml
Accept-Language: pt-BR,pt,en-US,en
Accept-Encoding: gzip, deflate
Connection: keep-alive
```

Resposta

```
HTTP/1.1 200 OK
Date: Tue, 17 Jun 2014 01:20:13 GMT
Server: gws
Location: https://www.google.com.br/
Last-Modified: Tue, 17 Jun 2014 01:20:13 GMT
Content-Encoding: gzip
Content-Length: 234
Connection: closeContent-Type: text/html
```

<html>todo o html da página</html> *

Verbos:

GET, POST, PUT, DELETE



Saiba mais: [HTTP Status Codes](#)

Verbos HTTP(s)

- GET
- HEAD
- POST
- PUT
- DELETE
- CONNECT
- OPTIONS
- TRACE
- PATCH



Mão na massa

- Checar versão instalada com o comando: “node -v”
 - ◆ Versão recomendada V8.11.3
- Executar no prompt de comando:

```
C:\Users\vntbosa>node
> console.log('Hello World')
Hello World
undefined
> █
```

Exercício #1

- Criar um servidor simples com apenas uma rota:
 - ◆ GET /

Alguma dúvida?



Criação e Inicialização do Projeto

```
vntjoig$ mkdir node4tech  
vntjoig$ cd node4tech/  
vntjoig$ npm init -y
```

```
Wrote to /Users/vntjoig/Documents/Venturus/node4tech/package.json:  
  
{  
  "name": "node4tech",  
  "version": "1.0.0",  
  "description": "",  
  "main": "index.js",  
  "scripts": {  
    "test": "echo \\" Error: no test specified \\ && exit 1"  
  },  
  "keywords": [],  
  "author": "",  
  "license": "ISC"  
}
```

Ou digite apenas *npm init* para ir configurando ponto a ponto...

VENTURUS4TECH





O que é NPM?

- Gerenciador de pacotes do NodeJS
- Gerência dependências do projeto
- Repositório online de pacotes bibliotecas de código aberto
- Geralmente instalado junto com o Node.js



Saiba mais: [Documentação oficial do NPM](#)



Mãos na massa

- Checar versão instalada com o comando: “npm -v”
 - ◆ Versão recomendada V5.6.0



ES6

Algumas features...

VENTURUS4TECH

express



O que é express?

- É um framework web para NodeJS
- Simplifica a construção de APIs REST
- Baseado em rotas
 - ◆ **GET /users** (lista de usuários)
 - ◆ **GET /users/13579** (usuário id = 13579)



Saiba mais: [Site oficial do Express](#)



FERRAMENTAS: POSTMAN

- Faz requisições para REST APIs
- É possível simular uso de cabeçalhos e obter o retorno dos dados
- Permite a criação de APIs mock para uso na aplicação

Saiba mais: [Download Postman](#), [Testando sua API web com Postman](#)

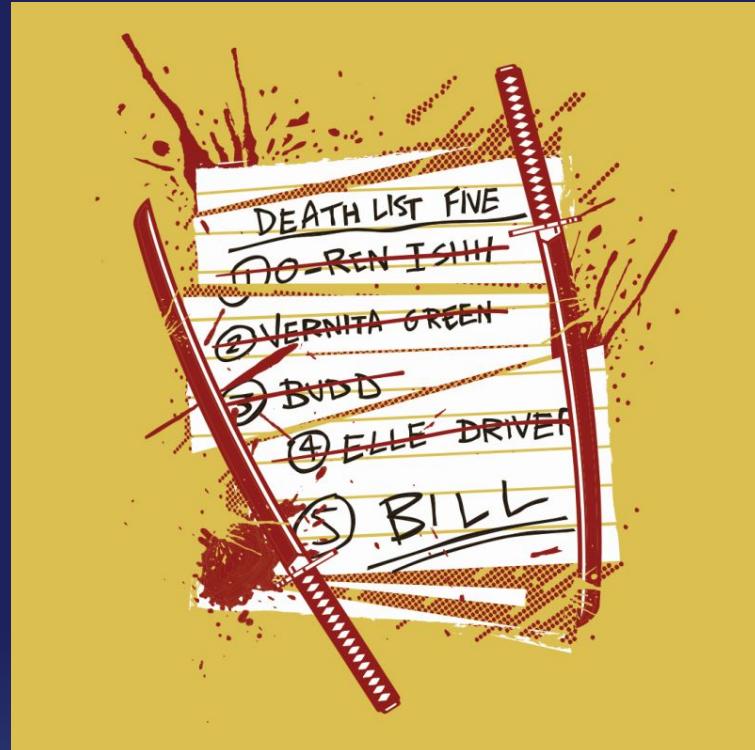
Exercício #2.1

- Vamos organizar a estrutura do nosso projeto para as próximas atividades



Modularização

- Dividir o arquivo app.js em arquivos menores, para facilitar leitura e manutenção
- Instalar o módulo consign com o npm
- Importar o módulo no app.js
- Importar as rotas com o consign



Saiba mais em: [Consign](#)

Exercício #2.2

- Vamos criar um servidor simples com Express utilizando REST com as rotas:
 - ◆ **GET /usuarios**
 - ◆ **POST /usuarios**

Jason



VENTURUS4TECH

JSON



O que é?

- JavaScript Object Notation
- É um formato de dados
- Não é dependente de linguagem
- É de fácil leitura e compreensão

Por que usar?

- Facilita a comunicação com o servidor.
- É fácil para as máquinas analisarem e gerarem.
- O JavaScript tem uma função incorporada para converter uma string, escrita em formato JSON, em objetos JavaScript nativos: `JSON.parse ()`
- É simples representar informações: para cada valor representado, atribui-se um nome (ou rótulo) que descreve o seu significado



Saiba mais: [JSON \(JavaScript Object Notation - Notação de Objetos JavaScript\)](#)



Let's try!

Exercício #3

- Usar servidor do Exercício 2 como referência para desenvolver servidor do VAGAS
- Rotas
 - ◆ Vagas
 - **GET /vagas**
 - **POST /vagas**



Saiba mais: [API Reference - 4.x](#)



Desafio #4

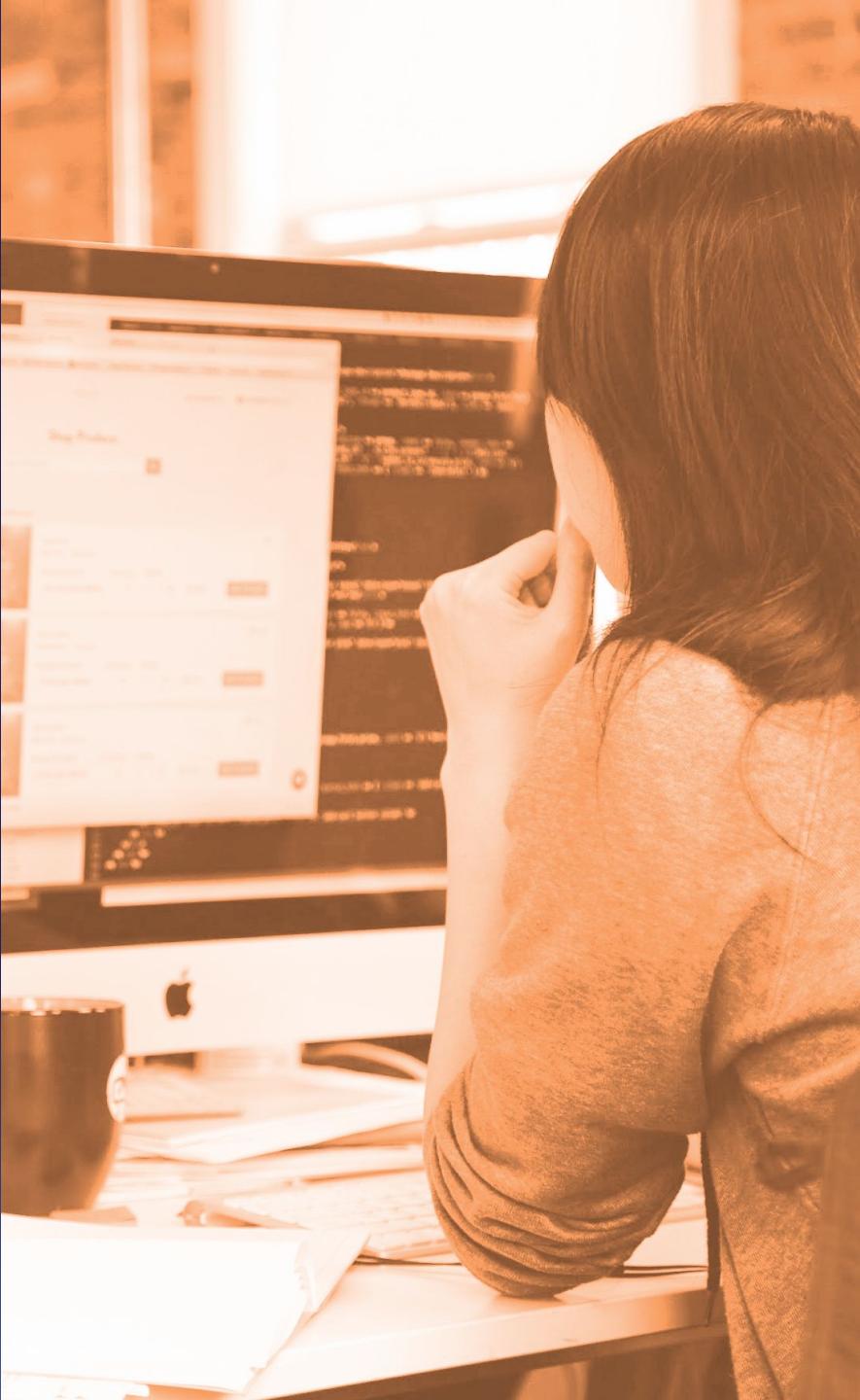
- Desenvolver rotas para:
 - ◆ GET por id
 - ◆ PUT
 - ◆ DELETE
- Validar informações no POST e PUT
 - ◆ Se quiser, utilize o express-validator
- Tratar possíveis erros com TRY/CATCH



Saiba mais em: [Express validator docs](#)



Saiba mais em: [HTTP Status Codes](#)



+Links e referências

- Se tiver dúvidas em Node ou quer estudar um pouco além:
 - [W3Schools](#)
 - [What exactly is Node.js?](#)
 - [API/REST/RESTful](#)
 - [Material para estudo](#)

Valeu galera!
Até amanhã!





Missão #DIA3 concluída!

